2-10-05 AF/2152

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re application of: Lyle Bate	§	Attorney Docket No.: 26530.4 (IDR-338)
Serial No.: 09/450,867	§	Customer No.: 27683
Filed: November 30, 1999	§	Group Art Unit: 2152
For: CACHING AND ACCESSING RIGHTS IN A DISTRIBUTED COMPUTING SYSTEM	§	Examiner: Willett, Stephan F.

REPLY BRIEF ON APPEAL

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

This Reply Brief is submitted in response to the Examiner's Answer dated December 9, 2004. For reference purposes, the latest list of claims is attached hereto as Appendix A.

GROUPING OF CLAIMS

Applicants believe that the Examiner's grouping of the claims is incorrect. For example, in response to the Examiner's assertion that claim 11's limitations are included in claim 1, claim 11 includes, inter alia, the following limitations: "means for accessing, by the agent, the first set of access rights of the principal to the resource; means for updating, by the agent, the first set of access rights to an access control list cache, wherein the access control list cache is located on a sixth one of the computers; means for receiving, at the access control list cache, a request from the principal for the first set of access rights; means for retrieving, by the access control list cache, the first set of access rights; means for forwarding, to the principal, the first set of access rights; and means for providing, to the principal, a deputization certificate adapted for enabling the

principle to copy one or more of the principal's access rights to at least one software entity." Those limitations are not included in claims 1, 15, 23 or 29.

With respect to the remaining arguments presented by the Examiner, Applicants respectfully submit that since the Examiner merely offered broad assertions without providing any support, the Examiner failed to counter the arguments presented in the Applicants' Brief. Therefore, Applicants believe that for purposes of this appeal, the grouping of claims should be as follows:

As to the rejection of claims 1-7 and 10, the rejected claims stand or fall together.

As to the rejection of claims 11-14, the rejected claims stand or fall together.

As to the rejection of claims 15-18, the rejected claims stand or fall together.

As to the rejection of claims 23-28, the rejected claims stand or fall together.

As to the rejection of claims 29-38, the rejected claims stand or fall together.

ARGUMENT

A. REJECTIONS UNDER 35 U.S.C. § 112

Regarding the rejections of claims 1 and 34 under 35 U.S.C. § 112, the Examiner simply recited the previous rejections from the Final Office Action, and failed to respond to Applicants' argument that the use of "principal" to represent "software principal" is sufficiently clear in the context of the claim, as that is the only "principal" contained in claims 1 and 34. Therefore, Applicants respectfully submit that the rejections should be withdrawn.

Regarding the rejection of claim 34 under 35 U.S.C. § 112 and in response to the Examiner's response number 15 (page 9 of the Examiner's Answer dated December 9, 2004), Applicants respectfully submit that "the principal is terminated" is clear and supported by page 15, line 24 to page 16, line 3 of the specification. It is clear from Fig. 3 and the above-cited specification that "no longer present on the system" means "terminated." Specifically, Fig. 3 illustrates that once the deputy 302 is "terminated," it is no longer present on the system. Therefore, Applicants respectfully submit that the rejection should be withdrawn.

B. REJECTIONS UNDER 35 U.S.C. § 103

Since the Examiner simply recited previous rejections from the Final Office Action (with minor variations) with respect to all the pending claims, Applicants respectfully submit that the arguments presented in the Applicants' Brief are sufficient to overcome those rejections.

C. EXAMINER'S RESPONSE TO APPLICANTS' BRIEF

1. Number 15 Of The Examiner's Response (Page 10 Of The Examiner's Answer Dated December 9, 2004)

Again, Applicants respectfully submit that "the principal is terminated" is clear and supported by page 15, line 24 to page 16, line 3 of the specification. It is clear from Fig. 3 and the above-cited specification that "no longer present on the system" means "terminated." Specifically, Fig. 3 illustrates that once the deputy 302 is "terminated," it is no longer present on the system. Therefore, Applicants respectfully submit that the rejection should be withdrawn.

2. Number 16 Of The Examiner's Response (Page 10 Of The Examiner's Answer Dated December 9, 2004)

Contrary to the Examiner's assertion that "user in these arts includes client and/or any subprocessing clients . . .," U.S. Patent No. 6,178,510 to O'Connor et al. ("O'Connor") explicitly recites the following: **The term "user" may include** multiple **persons** that meet the access criteria and that share a communication terminal. (col. 7, line 67 – col. 8, line 2) (emphasis added). Therefore, Applicants respectfully submit that the Examiner's response directly conflicts with the cited reference and should be disregarded.

3. Number 17 Of The Examiner's Response (Page 10 Of The Examiner's Answer Dated December 9, 2004)

The Examiner's response mischaracterized Applicants' arguments. Firstly, in the response dated August 4, 2003 (referred to as Paper No. 10 in the Examiner's Answer), Applicants argued that "the cited text of O'Conner fails to teach or suggest [the] deputization." (page 8, first line of the last paragraph) (emphasis added)

Secondly, in the response dated March 15, 2004 (referred to as Paper No. 14 in the Examiner's Answer), Applicants argued the following:

> Therefore, O'Conner simply discloses that the host acts on the user's instructions. It does not describe the delegation of rights to enable a software entity to access a resource using the delegated access rights without requiring intervention or control by the principal. (page 10, paragraph 5)

Therefore, Applicants respectfully submit that the arguments in the respective responses do not conflict with each other. Further, in the Examiner's Answer, the cited text of O'Connor fails to teach or suggest the specific ways of delegating rights as recited by each of the claims 11, 15, 23 and 29. Therefore, the Examiner's response failed to counter the arguments presented in the Applicants' Brief.

4. Number 19 Of The Examiner's Response (Page 13 Of The Examiner's Answer Dated December 9, 2004)

Here, the Examiner argued that the limitations in claim 1 were obvious, but failed to present any supporting references. Further, contrary to the Examiner's focus on "updating" only, Applicants respectfully submit that the referenced limitation in claim 1 includes: "**updating, by the agent, the access rights in an access control list cache**, wherein the access control list cache is coupled to the deputization point and to the principal[.]" (emphasis added) Therefore, the Examiner's response failed to counter the arguments presented in the Applicants' Brief.

5. Number 20 Of The Examiner's Response (Page 13 Of The Examiner's Answer Dated December 9, 2004)

The Examiner's response simply stated his personal opinion without presenting any supporting references. Thus his response is unpersuasive. Applicants also wish to submit that contrary to the Examiner's assertion, Fig. 3 and related text of the specification discloses the claim limitation.

6. Number 21 Of The Examiner's Response (Page 14 Of The Examiner's Answer Dated December 9, 2004)

The Examiner's broad statements failed to counter our arguments on pages 9-12 of the Applicants' Brief. Further, the cited text of U.S. Patent No. 6,157,953 to Chang et al. ("Chang") is as follows:

> multiple service hosts must re-authenticate and pass the administrator's credentials to each service host to which the administrator logs on. This is true since the administrator[.] (col. 12, lines 8-10)

In contrast, claim 11 requires "a deputization certificate adapted **for enabling the principle to copy . . . access rights** . . . ." (emphasis added) Therefore, Applicants do not believe that the cited text even remotely resembles the verbatim verbiage asserted by the Examiner.

Accordingly, Examiner failed to counter our arguments on pages 9-12 of the Applicants' Brief.

7. Number 22 of the Examiner's Response (page 14 of the Examiner's Answer Dated December 9, 2004)

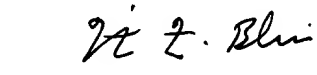
The Examiner's response simply recited statements presented in the Final Office Action, and thus failed to counter our arguments presented in the Applicants' Brief.

Conclusion

Accordingly, it is respectfully submitted that neither O'Connor nor Chang teaches or suggests the subject matter of claims 1-7, 10-18 and 23-38. Moreover, it is respectfully submitted that it is improper to combine the references because there is no motivation or suggestion for such combination to achieve the Applicants' claimed elements.

For all of the foregoing reasons, it is respectfully submitted that claims 1-7, 10-18 and 23-38 be allowed. A prompt notice to that effect is earnestly solicited.

Respectfully submitted,



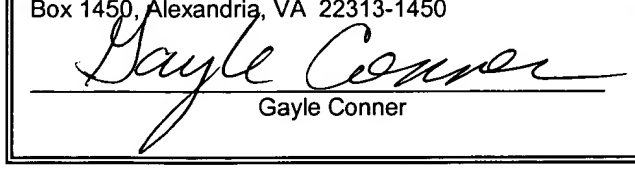
Timothy F. Bliss
Registration No. 50,925
Attorney for Applicants

Date: February 9, 2005
Haynes and Boone, LLP
901 Main Street, Suite 3100
Dallas, TX 75202-3789
(972) 739-8638

EXPRESS MAIL NO.:EV 334577467 US

DATE OF DEPOSIT: February 9, 2005

This paper and fee are being deposited with the U.S. Postal Service Express Mail Post Office to Addressee service under 37 CFR §1.10 on the date indicated above and is addressed to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450



Gayle Conner

APPENDIX A

1. A method for caching and accessing access rights to at least one resource in a distributed computing system, the method comprising:

accessing, by a software agent, a directory service, wherein the agent is located on a deputization point coupled to the directory service, and wherein the directory service comprises the access rights of a software principal to a resource;

updating, by the agent, the access rights in an access control list cache, wherein the access control list cache is coupled to the deputization point and to the principal;

receiving, at the access control list cache, a request from the principal for the access rights stored in the access control list cache;

retrieving, from the access control list cache, the access rights;

forwarding, to the principal, the access rights;

delegating one or more of the principal's access rights to at least one software entity; and

accessing the resource, by the software entity, using the delegated access rights without requiring intervention of the principal to authenticate access requests by the software entity, wherein tasks can be accomplished by the software entity without control by the principal.

2. The method of claim 1, wherein the access control list cache is comprised of a first table comprising the principal that has access to the resource.

3. The method of claim 1, wherein the access control list cache is comprised of a second table comprising the access rights of the principal to the resource.

4. The method of claim 1, wherein the access control list cache is comprised of a third table comprising a cached access to the resource object.

5. The method of claim 2 further comprising invoking, by the directory service, a resource manager, if the first table does not contain the principal that has access to the resource, wherein the resource manager is coupled to the directory

service and comprises access information and access rights of the principal to the resource.

6. The method of claim 5 further comprising mapping, by the resource manager, an access control of the access rights in the resource manager to an access control of the rights in the directory service.

7. The method of claim 6 further comprising updating, by the resource manager, the mapped access control of the access rights to the access control list cache.

8 and 9 (Cancelled).

10. The method of claim 1, further comprising at least one of the following actions from the group consisting of:

asynchronously updating, by the agent to the access control list cache, the access rights, when the access rights are added to the directory service;

asynchronously updating, by the agent to the access control list cache, the access rights, when the access rights are removed from the directory service;

asynchronously updating, by the agent to the access control list cache, the access rights, when the request from the principal is received;

synchronously updating, by the agent to the access control list cache, the access rights, when the access rights are added to the directory service;

synchronously updating, by the agent to the access control list cache, the access rights, when the access rights are removed from the directory service;

synchronously updating, by the agent to the access control list cache, the access rights, when the request from the principal is received;

updating, at a scheduled time, the access rights by the agent to the access control list cache; and

updating, after a time to live has expired, the access rights by the agent to the access control list cache.

11. A distributed computing system supporting access control caching, the system comprises:

a plurality of computers, each having a memory and a processor;

a plurality of communication links connecting the plurality of computers;

a principal located on a first one of the computers;

an agent located on a second one of the computers;

a resource located on a third one of the computers;

a first set of access rights located on a fourth one of the computers;

a second set of access rights located on a fifth one of the computers;

means for accessing, by the agent, the first set of access rights of the principal to the resource;

means for updating, by the agent, the first set of access rights to an access control list cache, wherein the access control list cache is located on a sixth one of the computers;

means for receiving, at the access control list cache, a request from the principal for the first set of access rights;

means for retrieving, by the access control list cache, the first set of access rights;

means for forwarding, to the principal, the first set of access rights; and

means for providing, to the principal, a deputization certificate adapted for enabling the principle to copy one or more of the principal's access rights to at least one software entity.

12. The system of claim 11 further comprises means for invoking the second set of access rights, if the first set of access rights is not located on the fourth one of the computers.

13. The system of claim 12 further comprises means for mapping an access control of the second set of access rights to an access control of the first set of access rights.

14. The system of claim 13 further comprises, means for updating the access control list cache with the mapped access control of the first set of access rights.

15. A computer storage medium having a configuration that represents data and instructions which will cause performance of method steps for caching and accessing access rights in a distributed computing system, the method comprising:

accessing, by a software agent, a directory service, wherein the agent is located on a deputization point coupled to the directory service having the access rights of at least one principal to at least one resource;

updating, by the agent, the access rights to an access control list cache, wherein the access control list cache is coupled to the deputization point, and wherein the access control list cache is coupled to the principal;

receiving, at the access control list cache, a request from the principal for the access rights;

retrieving, by the access control list cache, the access rights;

forwarding, to the principal, the access rights;

forwarding, to the principal, a deputization credential empowering the principal to deputize software entities; and

deputizing, by the principal, at least one of the software entities, wherein the software entity can exercise one or more of the principal's access rights due to the deputization.

16. The configured storage medium of claim 15 further comprising invoking, by the directory service, a resource manager, if the access control list cache does not contain one of the access rights, wherein the resource manager is coupled to the directory service, and wherein the resource manager comprises the one right.

17. The configured storage medium of claim 16 further comprising mapping, by the resource manager, an access control of the one right to an access control of the access rights.

18. The configured storage medium of claim 17 further comprising updating, by the resource manager, the mapped access control of the access rights to the access control list cache.

19-22 (Cancelled).

23. A method for controlling access within a computer system using deputization, the method comprising:

receiving an access authorization request at a deputization point from a principal, wherein the access authorization request requests validation of the principal's identity;

determining whether to validate the principal based on the access authorization request;

identifying one or more resource access permissions for the principal if the principal is validated, wherein the resource access permissions enable the principal to access one or more resources; and

providing the principal with deputizing authority at the identified access authorization level, wherein the deputizing authority comprises a deputization credential that enables the principal to give at least one software entity within the computer system a level of resource access permission equal to or lesser than the principal's resource access permissions.

24. The method of claim 23 wherein determining whether to validate the principal includes comparing information present in the access authorization request to a plurality of access rights contained in an access control list cache.

25. (Previously presented): The method of claim 24 further comprising:

invoking a resource manager if the access control list cache does not contain an access right associated with the access authorization request;

locating the access right associated with the access authorization request; and

mapping the access right into the plurality of access rights.

26. The method of claim 23 further comprising deputizing, by the principal, a first software entity, wherein the first software entity has a level of resource access permission equal to or lesser than the principal's resource access permissions.

27. The method of claim 26 wherein deputizing includes defining a lifespan of the deputization.

28. The method of claim 26 further comprising deputizing, by the first software entity, a second software entity, wherein the second software entity has a level of resource access permission equal to or lesser than the first software entity's level of resource access permission.

29. A computer-executable method for delegating permission from a software principal to a software deputy within a computer network to access at least one resource that is accessible to the principal, the method comprising:

receiving a request from the principal for a deputy credential, wherein the request includes the principal's identity and at least one permission to be assigned to the deputy;

sending the deputy credential to the principal, wherein the deputy credential enables the principal to assign the permission to the resource to the deputy;

receiving a deputization request from the principal to assign the permission to the deputy; and

assigning the permission to the deputy, wherein the deputy can independently access the resource using the assigned permission without being controlled by the principal.

30. The method of claim 29 further comprising imposing a lifespan on the assignment of the permission, wherein the assignment will expire at the end of the lifespan.

31. The method of claim 29 further comprising imposing a lifespan on the deputy, wherein the deputy will terminate at the end of the lifespan.

32. The method of claim 29 further comprising:

determining if a deputy identified in the deputization exists; and

creating the deputy if the deputy does not exist.

33. The method of claim 32 further comprising identifying a start time in the deputization request for assigning the permission to the deputy, wherein the permission is not assigned to the deputy until the start time.

34. The method of claim 33 wherein the principal is terminated in the computer network prior to the start time.

35. The method of claim 29 further comprising verifying that the principal is permitted to access the resource prior to sending the deputy credential to the principal.

36. The method of claim 29 wherein the deputy is in a namespace that is not accessible to the principal, and wherein the deputy can use the permission to access a resource in the namespace that is not accessible to the principal.

37. The method of claim 29 wherein the request from the principal for a deputy credential includes a plurality of permissions to be assigned to the deputy, and wherein the deputy credential sent to the principal permits the principal to assign only a portion of the plurality of permissions to the deputy.

38. The method of claim 29 further comprising

receiving a second request from the principal for a second deputy credential, wherein the request includes the principal's identity and at least a second permission to be assigned to the deputy;

sending the second deputy credential to the principal; and

assigning the second permission contained in the second deputy credential to the deputy, wherein the deputy includes permissions from both the deputy credential and the second deputy credential.